BLM LIBRARY
88050199

conceptual structure

Bureau Resource Information Management System (BRIMS)

Volume II – Reference Manual

U.S. DEPARTMENT OF THE INTERIOR
BUREAU OF LAND MANAGEMENT

BLM LIBRARY
SC-653, BLDG. 50
DENVER FEDERAL CENTER
P. O. BOX 25047
DENVER, CO 80225-0047

QA
76.8
.B85
C663
1977
v.2
c.2

BLM LIBRARY
RS 150A BLDG. 50
DENVER FEDERAL CENTER
P.O. BOX 25047
DENVER, CO 80225

ID88050199

QA
76.8
.B85
C663
1977
v.2
c.2

CONCEPTUAL SYSTEM STRUCTURE

FOR

BLM RESOURCE INFORMATION MANAGEMENT SYSTEM

REFERENCE MANUAL

FIRST DRAFT

10-20-77

BLM LIBRARY
RS 150A BLDG. 50
DENVER FEDERAL CENTER
P.O. BOX 25047
DENVER, CO 80225

PREFACE

This Reference Manual is the second volume of the Conceptual System Structure. It contains task descriptions and a glossary. It provides more detailed information about tasks and terms than that contained in the first volume - System Summary.

Task descriptions are sequenced within functions and functional areas. Task names and alphabetic character ID's are identical to those presented in the System Summary.

Terms within the Glossary are in alphabetical sequence.

TABLE OF CONTENTS

Page #

1.0 DATA ENTRY 1
1.1 Data Entry Supervisor 2
1.2 Input Device Interface 6

2.0 DATA PRESENTATION 8
2.1 Data Presentation Supervisor 9
2.2 Alphanumeric Paging and Presentation Services 11
2.3 Graphic Presentation Services 14
2.4 Output Device Interface 19

3.0 USER INTERFACE 21
3.1 User Interface Supervisor 22
3.2 System Security 24
3.3 User Assistance 26
3.4 Transaction Analysis 30
3.5 Transaction Queueing 32
3.6 Output Message Routing 33
3.7 Error Handling 35

4.0 APPLICATION INTERFACE 37
4.1 Application Supervisor 38
4.2 Application Memory Management 40

TABLE OF CONTENTS (continued)

Page #

5.0 DATA BASE SERVICES 41

5.1 Data Storage and Maintenance 42

5.2 Data Retrieval 45

5.3 Data Base Manager Interface 48

5.4 Data Base Manager 49

6.0 APPLICATION ROUTINES 50

7.0 COMMON SERVICES 52

7.1 Alphanumeric Output Processing 53

7.2 Graphic Output Processing 56

7.3 Business Graphs and Charts 59

7.4 Spatial Processing 60

7.5 Mathematics and Statistics 69

8.0 SYSTEM SUPPORT 70

8.1 Data Base Support 71

8.2 Telecommunication Support 73

8.3 General Activity 74

9.0 GLOSSARY 76

SECTION 1

DATA ENTRY

1.1 DATA ENTRY SUPERVISOR

A. Obtain data from Input Device Interface

This involves asking for and/or accepting data from the Input Device Interface. The data will have been converted to internal character code.

B. Translate graphic device coordinates to internal system coordinates

The programs which perform this task use a translation table to convert device coordinates from a digitizer to the internal coordinate system. The translation process compensates for:

- The position and angle of the document which has been placed on the digitizer
- The scale of the document
- Any linear distortion of the document such as expansion or contraction which may have occurred during a reproduction process

The translation table is created by a registration procedure at the time the document is placed on the digitizer. The registration procedure is described in a section under Spatial Processing.

Translation of cursor position on the screen to the internal coordinate system is a slightly different problem. The translation parameters change each time the image is zoomed or the viewable window is moved.

C. Identify menu and/or function key selection

The programs which perform this task handle:

- Hardware function keyboards on alphanumeric or graphic terminals
- Screen menus on alphanumeric or graphic displays. This includes any menu entries that simulate hardware function keys.
- Paper menus on digitizer boards or tables. This includes any alphanumeric keyboards or simulated function keys which appear on the menu.

If the source of the selection is a paper menu on a digitizer, the programs determine whether or not the devise coordinates are within the area of the digitizer where the menu has been placed.

For all sources, the programs determine which entry has been selected and insert a standard internal identifier into the input message.

D. Arrange data in a standard input format

This task contains the following sub-tasks:

- Collect data from the various devices at a work station. A work station may consist of a digitizer, an alphanumeric screen, a graphic screen, a hardware keyboard and a function keyboard. The operator may inter-mix entries from the various devices. This sub-task consists of collecting and arranging these entries into a single message just as if they were entered from a single device.
- Create a standard input message. This task consists of arranging data from the various types of work stations into a common standard input form. Subsequent programs should not even be aware of the type of input device used to enter the message.

E. Pass data to receiving routine

For buffered input devices (normally alphanumeric devices), this consists of passing a buffer load at a time. For non-buffered devices (normally graphic devices), this consists of passing entries, such as translated digitizer coordinates, one at a time.

F. Detect errors and take appropriate action

All programs, which perform Data Entry Supervisor tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

1.2 INPUT DEVICE INTERFACE

A. Obtain data from the input device

Some of the characteristics of data entry which may vary with the device type are:

- . Content, e.g., an entire screen or only the changes
- . Method, e.g., one character at a time or a string of characters from an input buffer
- . Character Codes, e.g., EBCDIC or ASCII
- . Initiation, e.g., external interrupt or periodic polling.

The programs which perform this task for a device takes the unique characteristics of the device into consideration and obtains the input data in a manner most suitable for subsequent processing.

B. Translate device character code to internal character code

This task involves translating the bit configuration used to represent a character on the device to the bit configuration used to represent the same character within the computer system.

C. Pass data to the Data Entry Supervisor

This task involves passing data to the Data Entry Supervisor after it has been translated to the internal character code.

D. Detect errors and take appropriate action

All programs, which perform Input Device Interface tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

SECTION 2

DATA PRESENTATION

2.1 DATA PRESENTATION SUPERVISOR

A. Obtain output data

This involves reading the device independent Output Buffer or at least the header portion of the buffer.

B. Determine the destination and device type

This involves examination of the header portion of the output buffer to determine the destination of the output. It also includes checking a table to determine the type of device at that location.

C. Pass data and control to the appropriate presentation service or device interface

This consists of invoking one of the following and passing the required parameters and/or data.

- Alphanumeric paging and presentation services
- Graphic presentation services
- Output device interface

D. Process special output messages

Some examples of special output messages are:

- "Alert" messages created by Output Message Routing to inform a terminal or user about the availability of output.
- "Alert" messages entered from a special terminal to inform users that the system will be down between certain hours.
- Feedback to a digitizer operator.
- Operator instructions such as:
  - mount a special form
  - mount a tape
- Etc.

Normally, this task will only consist of invoking the appropriate Output Device Interface and passing the message.

E. Detect errors and take appropriate action

All programs, which perform Data Presentation Supervisor tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

2.2 ALPHANUMERIC PAGING AND PRESENTATION SERVICES

A. Format output into device dependent pages

This task consists of breaking the device-independent output into pages which will fit on the destination device. For some reports and some devices, there will be nothing to do. Some can be handled by breaking a single line into two lines and identing the fields of the second line. Some reports for a specific device may be so complex that a pre-defined layout described in an output spec table is required. Some reports for a specific device may be so complex that specially written programs are required. Some output devices, such as tapes or card punches, have no requirement for pages like those required for display screens or printers. For these types of devices (tapes and card punches), a record or card image is treated as a page.

B. Present pages on a display device in any user desired sequence

This task consists of the following sub-tasks:

- Store pages in the page queue. This includes establishing and maintaining any required indexes or pointers. It also includes presentation of the first page of the report.

- Process request for a page. Some sample requests are: next page, previous page, last page, or a specific page number.
- Process request to leave a current report in the page queue. Normally when a user has been presented with the last page of a report and his next transaction produces a new report, the new report replaces the previous report in the page queue. This sub-task causes the normal procedure to be bypassed so that the user can execute another transaction.
- Then resume the viewing of the current report.
- Provide a list of reports and pages in the page queue.
- Delete reports from the page queue. This task may be executed upon user request or upon user sign-off.

C. Present pages to any sequential device in sequential order

This task involves accepting the device dependent pages from task A and passing them to the Output Device Interface. It may or may not involve temporarily storing the pages in a page queue.

D. Pass data to the Output Device Interface

This consists of invoking the appropriate device interface and passing the required parameters and/or data.

E. Detect errors and take appropriate action

All programs which perform alphanumeric paging and presentation tasks perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

2.3 GRAPHIC PRESENTATION SERVICES

A. Obtain graphic output

This task consists of reading the device independent graphic Output Buffer.

B. Determine the portion of the Output Buffer to be presented

This task is executed upon any of the following conditions:

- The processing of the original display request
- The processing of a "zoom" request
- The processing of a "move window" request
- The processing of a "plot" request

The task involves determining whether all or just a portion of the Output Buffer is to be presented. If just a portion is to be presented, it includes determining the window size in terms of the internal system coordinates.

C. Determine the size of the device window

This task involves determining whether all or just a portion of the destination device is to be used for presentation of the image. In some cases, portions of a screen are reserved for text including: title, legend, error messages, notes, or attribute information. For plotters, a specific size document may be requested which is less than the maximum size. This task includes determining the device coordinates in device units of the device window to be used for presentation of the image.

D. Determine translation parameters between internal coordinates and device coordinates

This task involves determining the parameters required to translate the internal system coordinates (for the portion of the Output Buffer to be presented) to the device coordinates (for the portion of the device to be used for presentation of the image). The actual translation of the coordinates are performed by the Output Device Interface using the parameters determined by this task.

E. Clip the portion of characters and vectors not in the device window

The programs which perform this task have the responsibility to remove the portion of a graphic image which will not be visible on the specified device at the specified scale. This process includes removing pieces of a character or a vector. This capability is required when the desired graphic window is just a portion of the Output Buffer rather than the entire images. The execution of this task is required only for those specific devices in which this capability is not provided by the vendor as hardware or graphic primitive features.

F. Manipulate the graphic image

This task includes the following image manipulations sub-tasks which do not require a change to the device-independent graphic Output Buffer.

- Move window
- Zoom
- Turn-off or turn-on a graphic layer

These sub-tasks are described in the following paragraphs.

Move window

The programs which perform this task have the responsibility to re-position the viewable window to a different portion of the entire available image. The size or scale of the viewable window is not changed. The options available under this task are:

- Move the window so that a specified point is in the center of the screen
- Move the window in a specified direction. The optional directions are the standard eight compass directions. The point which was on the edge of the screen (in the specified direction) is moved to the center of the screen.

Zoom

The programs which perform this task have the responsibility to enlarge or reduce the viewable size of the available image. All vectors, characters and symbols are enlarged or reduced in the same proportion. The center point of the screen does not change. The options available under this task are:

- Zoom to a viewable window size so that the entire available image is displayed
- Zoom to a specified viewable window size
- Zoom to one-half of the current viewable window size
- Zoom to twice the current viewable window size

Turn off or turn on a graphic layer

A graphic layer consists of a pre-defined set of entities. For example, a graphic layer could be defined as any one or any combination of the following: political boundaries, BLM administrative boundaries, cadastral grids, roads, rivers, etc. The programs which perform this task have the responsibility to remove or restore a graphic layer to a displayed image. Data is not actually removed from the available image. It is simply not used during the process of creating the display.

G. Break graphic output into contiguous sections

The programs which perform this task have the responsibility to break a map into sections so that they can be manually pasted or taped together to form a large map which has continuity at the edge of the section. This capability is required if the requested map cannot be produced on one piece of paper by the designated plotter at the specified scale.

H. Pass data to Output Device Interface

This task consists of invoking the appropriate Output Device Interface and passing the required parameters and/or data.

J. Detect errors and take appropriate action

All programs, which perform Graphic Presentation Services tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

2.4 OUTPUT DEVICE INTERFACE

A. Accept data from Alphanumeric Paging and Presentation Services, Graphics Presentation Services, or Data Presentation Supervisor

This task consists simply of accepting the output data and the associated parameters.

B. Convert the data to the specific device dependent form and format

The programs which perform this task have the responsibility to perform the following sub-tasks if necessary:

- Select and rearrange the content of the output data into the required device-dependent format.
- For graphic output, translate the internal system coordinates to device coordinates.
- Translate internal character codes to device character codes.
- Remove standard control codes and insert proper control codes that are unique to the output device. Some examples of control codes:
  - return carriage
  - start a new line
  - start a new page
  - etc.

C. Invoke access methods or graphic primitives

The programs which perform this task have the responsibility to invoke the vendor supplied and/or BLM provided device interface software. The access methods may vary depending upon the device class and type and also upon whether the device is channel connected or connected through a telecommunication line. The access methods and primitives cause the data to be actually presented to the output device.

D. Detect errors and take appropriate action

All programs, which perform Output Device Interface tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it cannot be corrected, a unique error code is assigned for each type of error.

SECTION 3

USER INTERFACE

3.1 USER INTERFACE SUPERVISOR

A. Control the execution sequence of the User Interface Functional Area programs

This task consists of invoking other User Interface routines and assuring that they have access to the required parameters and data.

B. Detect interrupts from a user or from the system to abort the processing of a transaction

This consists of:

- Detecting a user entered request to abort the processing of a specific transaction.
- Detecting and intercepting an abnormal job termination. This condition would not occur in a perfectly debugged system but that state is rarely, if ever, reached.

This task consists of taking whatever steps are necessary to abort the transaction.

C. Detect transactions that alter a normal processing sequence and resequence appropriately

Examples of transactions which could alter the processing sequence are:

- A request to re-route an output from one device to another
- The signing-off of a terminal user before the processing of his transaction has been completed

This task consists of detecting this type of transaction and taking whatever steps are necessary to alter the processing sequence.

D. Create the required job stream for processing "batched" transactions

This task consists of creating the data stream necessary to execute the batch job. It also includes either the initiation of the job or placement of the job in the system job queue. This task may require (depending upon the available system support) writing a message to the computer operator and asking him to actually initiate the job.

E. Determine error conditions and assign error codes

All programs, which perform User Interface Supervisor tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

3.2 SYSTEM SECURITY

A. Verify a user sign-on for: syntax, user identification, project identification, and password

This task includes the following sub-tasks:

- Check for correct spelling, correct punctuation, and for missing parameters.
- Check to assure that the terminal operator is authorized to use the system.
- Check for a valid project number. The project in this context is synonomous with "charge number". If project ID is not required for all terminal usage, then a default project number is used if appropiate.
- Check to assure that the password entered is a currently acceptable password.

B. Establish the specific system access allowed for each transaction based upon the sign-on

This involves checking tables to determine if the transaction identification code (TIC) which the user entered is an authorized transaction for that specific user. It does not include the checking of qualifiers or data. This additional check is performed by Data Base Services.

C. Identify and log attempts to execute unauthorized transactions

This includes the following sub-tasks:

- Identify the unauthorized transaction
- Create an entry in the unauthorized transaction log

- Send a message to a pre-defined terminal about the entry
- Return the transaction to the user with an explanation that the transaction was un-authorized

D. Log summary of authorized terminal usage activity

This involves inserting entries in a log which include items such as: Project ID, User ID, Terminal ID, date, Sign-on time, Sign-off time, etc.

E. Handle terminal sign-off

This involves creating an entry in the tables used to maintain terminal usage data. This task is executed automatically at sign-on time for a new user, if the previous user did not sign-off.

F. Determine error conditions and assign error codes

All programs, which perform System Security tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

3.3 USER ASSISTANCE

A. Provide assistance to users through a conversatioan exchange using menus, questions, index maps, and/or explanatory text as appropriate

This task consists of the following sub-tasks:

- Provide pre-formatted displays of forms and menus. This includes identifying the required display, retrieving the display and storing the display in an Output Buffer
- Provide lists of available data. This includes retrieving the following lists and storing them in an Output Buffer.
  - List of files and records per file
  - List of saved Work Spaces and Output Buffers.
  - List of data elements in a file
  - List of available data themes in the geographic area of interest
  - List of available data elements within a theme and within the geographic area of interest
  - List of keys in a file
- Provide explanation of displayed items. This includes retrieving the following descriptions and storing them in an Output Buffer.
  - Description of a data element and the valid entries or a valid range of entries
  - Description of a menu key or a function key and the corresponding system response if the key is selected

    - Description of valid sequences of menu selections
    - Description of items on an output report
    - Description of a graphic symbol
- Provide displays of index maps. This is required in some situations in order to help the user identify his geographic area of interest. This sub-task consists identifying the required index map, retrieving it, and storing it in an Output Buffer.

  Provide questions for the user to answer in order to define a transaction entry or to help identify the next system response.

B. Accumulate user entries into transaction format entry by entry

This task contains the following sub-tasks

- Obtain the user entry
- If the entry is a menu key or function key selection, replace the identifier with a pre-defined character string, if necessary.
- Determine the location in the transaction image where the entry is to be placed
- Insert the entry in the transaction image
- Detect the last entry for a transaction
- Make the completed transaction available for further processing

C. Perform preliminary edit and validation of transaction entries and re-prompt if invalid

This tasks consists of the following sub-tasks:

- Validate the sequence of entries
- Perform pre-defined precursory validity checks that have been defined for the specific transaction
- If an invalid entry is detected, provide a response message. This includes using some sub-tasks of task "A" to explain an entry and its acceptable value or range of values.

D. Provide feedback to terminal operator

This task consists of the following sub-tasks:

- Provide positive or negative feedback to digitizer operator

  The feedback could be of several forms:
- An audio beep
  - A single beep for positive feedback which indicates the entry was valid
  - A double beep for negative feedback to indicate an invalid entry
- A light or lights on the digitizing puck or stylus
  - A green light for positive feedback
  - A red light for negative feedback

  Normally, the digitizer operator will know what he did wrong after he receives the negative feedback signal. He can then proceed to make the correct entry. If he can not determine what he did wrong, he can ask for an explanation

- Provide user with an english translation of the transaction he has entered
- Provides user with transaction status feedback. An example of this is "PLEASE WAIT- YOUR TRANSACTION IS NOW BEING PROCESSED."

E. Insert implied or default entries into the transaction

An example of this is to insert the district where a terminal is located as the default for "Geographic area of interest. This task is applicable in User Assistance only when formatting the transaction entry by entry.

F. Determine error conditions and take appropriate action

All programs, which perform User Assistance tasks, perform checks to detect errors associated with their area of interest. If an error is detected in a user entry, a correction is attempted by prompting the user. A unique error code is assigned for each other type of error detected.

3.4 TRANSACTION ANALYSIS

A. Validate user-supplied transaction entries

This task normally applies to those transactions where the user has entered the entire transaction through an alphanumeric keyboard. This task consists of the following sub-tasks.

- Check for correct spelling, correct punctuation and for missing parameters
- Perform pre-defined precursory validity checks for the specific transaction

B. Insert implied or default entries into the transaction

This task normally applies to those transactions where the user has entered the entire transaction through an alpha-numeric keyboard. An example of this to insert the district where a terminal is located as the default for "Geographic area of interest."

C. Translate the transaction to a standard application format

This task consists of the following sub-tasks:

- Replace a macro type of transaction with a set of standard application transactions
- Divide and sequence the inputted transaction into a format convenient to the processing program

The purpose of this task is to translate input from a form that is convenient to the user to a form that is convenient for the processing program. This may require some additional sub-tasks such as "accumulate several transactions" in order to create the input in a form that is convenient to an application program.

D. Verify the reasonableness of the request

This involves development of a rough estimate of computer time required to process a transaction. This is required in the final system in order to question or reject unreasonable requests such as one which would require a record-by-record search of all data bases. Some examples of responses which could be made to unreasonable requests are:

- Are you sure that you really want to use 16 hours of computer time to process this transaction?
- Please try to define a smaller geographic area of interest.
- Please ask "User Assistance" for help in the formulation of a more reasonable request.

E. Determine error conditions and assign error codes

All programs, which perform Transaction Analysis tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

3.5 TRANSACTION QUEUEING

A. Determine the priority of each transaction

This tasks consists of using tables to find the priority of the specific transaction.

B. Queue batchable transactions for batch processing

This tasks consists of using tables to determine if the type of transaction is batchable. If so, it is inserted into the batch queue. The User Interface Supervisor will subsequently use the transaction during the creation of a batch job stream.

C. Insert on-line serially processable transactions into appropriate queue for initiation by Application Interface

This task consists of using the previously determined priority to determine if it should be processed on a first in-first out basis. If so, it is inserted into the appropriate queue

D. Insert priority transactions into appropriate priority queue.

This tasks consists of using the previously determined priority to determine if it should be processed immediately. If so, it is inserted into the appropriate priority queue.

E. Determine error conditions and assign error codes

All programs, which perform Transaction Queueing tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

3.6 OUTPUT MESSAGE ROUTING

A. Maintain message routing information

This task consists of the following sub-tasks:

- Insert a table entry for destination of output, if any, for each transaction being processed. The destination could be a specific terminal, a specific user or both.
- Insert an entry to indicat that output should be placed in a "hold file" under the following circumstances
  - If the transaction is batched
  - If the user signs off before processing has been completed and he did not abort the transaction.
  - If the user enters another transaction before the processing of the previous transaction has been completed
  - If the output is for a destination other than the paired device of the source of the transaction
  - If operator intervention is required.
- Insert an entry to locate output that has been placed in a "hold file"
- Insert an entry to indicate that processing of the transaction has been completed (including any necessary processing by Data Presentation)
- Remove the table entry when the processing is completed and the next transaction entry is not a "save" or "hard copy" request

B. Correlate output with specified destination(s)

This task consists of finding the table entry and the corresponding destination for the specific output.

C. Provide appropriate message to user regarding "staged" output

This consists of providing a message to the terminal or user that his output is ready and that he should enter a transaction to request the output. The message could be any one of several forms. A light on the terminal, an audio alarm or gong or a english-message written on the bottom line of the screen

D. Route "staged" output to requesting terminal for appropriate action by user.

This consists of identifying the requested output and making it available for Data Presentation to send. If operator intervention is required, the output may be made available in two sections. The first section could contain the operator instructions such as mount a special form. After he has mounted the form, he enters another transaction to request the second section which is the real output.

E. Determine error conditions and assign error codes

All programs, which perform Output Message Routing tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

3.7 ERROR HANDLING

A. Create meaningful and consistent error messages from error codes assigned by other programs within the system when an error condition is encountered. This task consists of retrieving a pre-defined error message for the specific error code. It may also include placement of some parameters, which accompanied the error code, into pre-specified locations within the message the error message is then used to replace the error code in the output report. Some parameters may be used to highlight certain fields in the report. The report is placed in an Output Buffer for processing by Data Presentation.

B. Formulate appropriate messages to all concerned terminals This task consists of the following sub-tasks:

- Create messages for system programmers. Some specific types of errors will need to be analyzed by system programmers in order to determine the cause of the error. This sub-task consists of creating error messages for the system programmers upon detection of these types of errors. The message sent to pre-specified terminals such as a systems programmers terminal and/or a master terminal.
- Create messages for the computer operator.

  The programs which perform the Error Handling also perform checks to detect errors. Some errors can be assigned a code and be processed in the same manner as other errors. Some errors, however can not be processed and will require operator intervention such as system re-start. This sub-task creates these types of error messages and writes them directly to computer operator console and/or the master terminal.

C. Maintain a log of all error codes and conditions processed This task consists of writing each error code and message on a log. The log will be subsequently used by another program to analyze the types of errors and print statistical reports. Analysis of the reports may lead to changes in user procedures, changes to programs, hardware repairs or replacement, or other changes external to the system which could reduce the number of errors.

SECTION 4

APPLICATION INTERFACE

4.1 APPLICATION SUPERVISOR

A. Accept transaction from User Interface Functional Area

Once a transaction has completed validation, analysis and queueing, control is passed to this task.

B. Establish and maintain the processing sequence required for the transaction, then pass control to the first program.

This task consists of using tables to determine the pre-defined processing sequence for the specific application.

C. Control the sequence of program execution

As each program completes processing, e.g. Data Storage and Maintenance; the application supervisor will check if the program executed successfully. If yes, then it will pass control to the next program in the sequence. If no, it will turn control over to the Error Handler.

D. Process user requests for termination of a transaction

A user may enter a transaction then decide that it is in error or that he doesn/t desire the result. This task will allow termination of certain types of transactions.

E. Supervise the modification of the availability of a transactions, if an error condition indicates a program problem

This task consists of the following sub-tasks

- Detect an abnormal termination which indicates that there is a bug in a program
- Invoke a sub-ordinate program which subsequently "turns-off" all transactions that require the malfunctioning

program. These transactions are then not available to any user until they are "turned-on" again via a special transaction from the master terminal or computer operator.

F. Determine error conditions and assign error codes

All programs, which perform application supervisor tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

G. Return control to User Interface when processing is complete

This task consists of passing control, parameters, and/or data User Interface Supervisor

4.2 APPLICATION MEMORY MANAGEMENT

A. Allocate application space for transaction processing as necessary

This task consists of using tables and perhaps performing some calculations to determine the amount of space required to process a transaction. It includes obtaining the required space and making it available to the processing programs.

B. Keep track of all memory and on-line storage utilization

This task consists of maintaining an index and/or pointers to all storage and its current status. The status includes whether the space is being used or not. If it is being used, it includes information about how it is being used. Examples of how it is being used are: Output Buffer, Work Space, saved Output Buffers, saved Work Spaces, tables, programs, etc.

C. Release space no longer required

This task consists of updating the status of a specific piece of space. It is changed from "being used" status to "available" status.

SECTION 5

DATA BASE SERVICES

5.1 DATA STORAGE AND MAINTENANCE

A. Check and validate transaction entries

This task consists of the following sub-tasks

- Check to assure that numeric parameters only contain numeric values and alphabetic parameters contains only alphabetic characters.
- Check parameters to assure that they are within a valid range of values or they are valid codes.
- Check parameters to assure that the relationships between the parameters are valid.
- Validate relationship of input data to existing data. This sub-task consists of looking at existing data in the data base to determine if the input data's relation to it is valid. For example, for a new lease on a piece of land, the check will verify that there isn't another active lease already on that piece of land.

B. Convert the long form for a data entry into its coded equivalent for the data entry

An example of this task is the conversion of state and county names to FIPS codes.

C. Convert data to desired precision or coordinate system

An example of this is to truncate any extra digits entered from a digitizer.

D. Format identification data into required "key" format

This task consists of arranging parameters into the sequence required for use as a key to access the data base.

E. Perform preliminary steps to change a record

This task consists of the following sub-tasks:

- Invoke the Data Base Manager Interface to retrieve the record which is to be changed
- Make changes to the appropriate data element

F. Re-arrange data into the format required by the data base format and structure

This task consists of arranging data elements into the record format used for storing data.

G. Perform required processes to calculate or re-calculate data elements

This task consists of the following sub-tasks:

- Use Mathematics and Statistics routines to calculate data elements such as "average" or "mean"
- Use Spatial Processing routines to calculate data elements such as "acreage"

This task includes re-calculation of data elements that are impacted by changes to other data elements.

H. Log the transaction for use as an audit trail

Some transactions may result in several updates to several data bases. An example of this is a transaction which requests that all changes currently in a user Work Space be used to update the Master Data Base. This task provides an audit trail so that, if necessary, a change to a data base can be traced back to the transaction which caused the change.

J. Pass control to appropriate add, or insert, change, or delete task within Data Base Manager Interface

This task consists of simply calling the Data Base Manager Interface and passing the required parameters.

K. Repeat the process as required until all input has been processed

This task consists of repeating previously described data storage maintenance tasks until all entries have been processes.

L. Detect errors

All programs, which perform Data Storage and Maintenance tasks, perform checks to detect errors associated with their area of responsibility.

M. Attempt to correct the error

Some errors may be corrected with additional or revised input data. This task consists of whatever sub-tasks are necessary to attempt the correction

N. Attempt recovery in order to reconstruct the data base should it have been changed based upon previous data in the transaction before the error was detected

This task consists of taking whatever steps are necessary to restore the data base to the exact same status that existed before the erroneous transaction was entered.

P. In case the error can not be corrected, assign error code

This task consists of assigning a unique error code to each different type of un-correctable error.

5.2 DATA RETRIEVAL

A. Verify the retrieval requests transaction entries

This task consists of the following sub-tasks

- Check for valid field name
- Check for valid operators ($=, <, >, \leq, \geq$, etc.)
- Check to assure that numeric parameters only contain numeric values and alphabetic parameters contains only alphabetic characters
- Check parameters to assure that they are within a valid range of values or they are valid codes
- Convert long form name data to the equivalent code if required
- Check for valid connectors (and, or)

B. Determine the specific data elements that must be retrieved

This task consists of checking data entries and/or tables to determine all data elements to be retrieved. It includes determining what records must be retrieved in order to obtain the data elements.

C. Group retrieval parameters by theme

This task consists of collecting the retrieval parameters used to access a single data base and record type. The task assumes that different themes will be stored in different data bases and that separate retrieval requests must be made for each record type.

D. Determine which field names are "keys" in the data base

Retrieval request parameters whose names are keys are used to retrieve the data element or record, and non-key retrieval request parameters are used to check the data's qualification after the retrieval.

E. Group retrieval request lines for efficient access to the data base by the Data Manager

There may be several ways to access the desired records. For example, primary index keys and/or several secondary index keys. This task consists of attempting to identify the way which will return the fewest number of records.

F. Pass control to the Data Base Manager Interface for the actual retrieval

This task consists of simply calling the Data Base Manager Interface and passing the required parameters.

G. Discard non-qualifying records by comparing data elements, that were not used as a key, with the retrieval qualifiers

This task consists of the following sub-tasks:

- Compare retrieval qualifiers that were not used as keys with retrieved data
- Discard non-qualifying records

H. Use mathematical routines for data qualification if necessary

This task consists of using Mathematics and Statistics routines to compute data elements and comparing the computed items with the retrieval qualifiers. Records are discarded if they do not meet the retrieval criteria.

J. Use Spatial routines, if necessary, for spatial qualification

An example of this task is to determine if a specific point is within the specified geographic area of interest. Records are discarded if they do not meet the retrieval criteria.

K. Extract required data elements from retrieved data records and arrange them into the required format

The data elements needed to satisfy the request are extracted from the retrieved record and moved into a user specified format if necessary.

L. Place data in the user Work Space

This is an interim storage space used for subsequent processing by other programs. It can be used to produce tabular reports, tabular displays, graphic displays, hard copy maps or can be used by an application program or can be modified by the user during an interactive session.

M. Repeat the tasks required to obtain the qualifying record from the data base and to place data in the user Work Space

This tasks consists of repeating other Data Retrieval tasks until all requested data has been retrieved, extracted and stored in the work space.

5.3 DATA BASE MANAGER INTERFACE

A. Convert to syntactically correct request for the specific Data Base Manager if necessary

This task consists of constructing the request to the Data Base Manager using the required syntax

B. Check user authorization to make the request

This task consists of the following sub-tasks:

- Check user authorization for retrieval of the specific data elements
- Check user authorization to perform inserts, deletes, or replaces of the specific data elements.

C. Pass control to the Data Base Manager

This task consists simply of calling the Data Base Manager and passing the required parameters.

D. Determine error conditions and/or attempts at unauthorized access and assign appropriate error codes

This task consists of the following sub-tasks

- Check status code returned from Data Base Manager to determine if the request was successfully executed.
- Check to assure that the value of all retrieved data elements are within the range of user authorized limits.
- Perform checks within each Data Base Manager Interface to detect errors within the assigned area of responsibility.
- Assign a unique error code to each different type of error detected.

5.4 DATA BASE MANAGER

A. Perform inserts, deletes, replacements, and retrievals

This task consists of making physical accesses to the Master Data Base, Work Spaces and Files

B. Maintain relationships of all data, both physical and logical, and maintain all primary and secondary indexes

This task consists of the physical update of all indexes and pointers related to the Master Data Base, Work Spaces and Files.

C. Return all data records or data elements specified in the retrieval request

This task involves returning the requested data to the Data Base Manager Interface

D. Return successful or unsuccessful status based upon the result of any request

This task consists of returning a parameter to the Data Base Manager Interface to indicate a successful completion of the request or the reason for an unsuccessful completion.

E. Manage data base space usage

This consists of managing the pre-allocated space. It includes re-using space where records have been deleted

F. Determine error conditions and take appropriate action

All programs, which perform Data Base Manager tasks, perform checks to detect errors associated with their area of responsibility. If an error is detected and can be corrected, the routines attempt to make the correction. If it can not be corrected, the status code is established to indicate the reason for the failure.

SECTION 6

APPLICATION ROUTINES

6.0 APPLICATION ROUTINES

The tasks identified in the System Summary for Application Routines were simply examples. No attempt is made to identify and describe all tasks within this function. The tasks generally fall into two categories.

- The system-oriented tasks that are unique to the Resource Information Management System
- The application oriented tasks that are unique or that are not generally usable by more than one of 15 application packages

SECTION 7

COMMON SERVICES

7.1 ALPHA-NUMERIC OUTPUT PROCESSING

A. Sort data into required order

The data in Work Space is sorted into the desired order if necessary.

B. Prepare report preface or "cover sheet"

Certain output products may require standard explanatory information for understanding of the output. These are pre-formatted and included as a part of the output specification, then output when the report is requested.

C. Insert report header information

This consists of inserting report or output titles, date, page numbers, column titles, etc.

D. Position data within the Output Buffer

This task consists of transforming the data from the stored format to the desired output format.

E. Insert special characters and punctuation

This task includes inserting characters such as: dollar signs, decimal points, parenthesis, plus or minus signs, etc.

F. Convert coded data to proper name equivalent

An example of this would be to convert a FIPS code to a county name.

G. Accumulate and format totals, subtotals, averages, sub-averages, counts and subcounts

This task is performed, if required, by using data elements identified in a table containing the specifications for the output report.

H. Perform arithmetic operations

This task consists of performing addition, subtraction, multiplication and division using data elements in the Work Space and/or items contained in the output specifications. The spec also contains the data elements on which the operation is performed and the type of operation.

J. Control output structure through logical specification and data characteristics

An example of this task is to skip a line or start a new page if a specific data element in this line has a different value from the previous page.

K. Prepare and format output summary or trailer information

An example of this would be grand totals, averages, etc.

L. Retrieve preformatted forms

This task consists of retrieving a saved Output Buffer for use in the following task.

M. Reformat input data or retrieved data into preformatted form layout

This task consists of the following sub-tasks

- Merge an input transaction, in which an error was detected, with the form used to enter the data. The re-constructed screen will be used to highlight an invalid entry and/or to explain an invalid entry with an error message.
- Merge retrieved data with a preformatted screen to support an alpha-numeric inquiry into the data base.

N. Determine error conditions and assign error codes

All programs, which perform User Interface Supervisor tasks, perform checks to detect errors associated with their area of responsibility. A unique error code is assigned for each type of error detected.

7.2 GRAPHIC OUTPUT PROCESSING

A. Obtain entity data from the graphic Work Space or from a calling program

This task consists of obtaining the data from which a graphic image is to be created.

B. Replace entities with a graphic symbol

This task is described via several examples:

- Replace a district boundary with vector strokes of a double width solid line
- Replace a resource boundary with vector strokes of a single width dashed line
- Replace a polygon with vector strokes of a cross-hatched area
- Replace a polygon boundary with a specified line type and offset a portion of the line from a coincident line so that both lines will be variable
- Replace a portion of an entity or portion of an entity with an arc

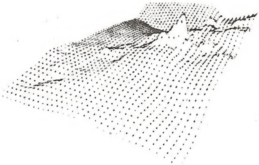
- Replace an entity or portion of an entity with a point-related symbol such as a circle, square, rectangle, triangle, arrow, cross, etc.

This task is performed by use of two tables: One table identifies the type of symbol to be used to represent each entity. There is a separate table of this type for each different output specification. Entities in the table can be over-ridden by user input which identifies the standard output specification and the desired deviations from the standard.

The second table is a symbol table and contains the vector strokes required for each different symbol.

C. Create a perspective view of a three dimensional surface

An example of this is shown below:



This task consists of creating the vector strokes required to represent the surface.

D. Use output specs to format an image

Some examples of this are:

1. Create a grid within a rectangular area
2. Create a legend to show the meaning of each symbol used in the image
3. Create a table box
4. Position alpha-numeric characters. This consists of creating the vector strokes to represent various characters at various fonts, sizes and angles. It includes breaking a line to represent a river and inserting the river name along the path of the removed section of the line. It includes inserting a highway name offset but in parallel with the highway. It includes positioning the name of a polygon at the centroid of the polygon. It includes the selection of a data element for an entity and positioning the characters to represent the value of the data element.
5. Break graphic symbols into graphic overlays or layers. The layers will normally represent themes such as rivers, or interstate highways that may turned-on or turned-off as a group to modify the graphic image.

E. Insert graphic symbols and their locations in a device independent form into the Output Buffer

This task consists of placing and formatting the vector strokes created by other Graphic Output Processing tasks into the Output Buffer.

7.3 BUSINESS GRAPHS AND CHARTS

The three tasks for this function are described in one general paragraph. The tasks are:

A. Create a bar chart from parameters and reference data

B. Create a line graph from two or more variables

C. Create a pie chart

These tasks consist of converting non-spatial data into a spatial format so that Graphic Output Processing can create a graphic representation of the data. They consist of creating special or pseudo-entities which can be represented by symbols such as lines, rectangles, arcs, and circles. The include the creation of text and its spatial position.

7.4 SPATIAL PROCESSING

A. Perform transformation calculations between external points and the internal coordinates system

This task consists of the following sub-tasks:

1. Calculate transformation parameters for digitized points from a tablet. These parameters are calculated during the process of registering a map on the tablet. It takes into consideration the position and angle of the map on the tablet. It uses a set of equivalency points to calculate the transformation parameter. The set of equivalency points consist of points on the map and the corresponding equivalent points in the Master Data Base or the corresponding equivalent points in the internal system coordinates.

2. Calculate transformation parameters for points identified on the graphic screen. This calculation requires the use of the internal system coordinates of the currently viewable area on the screen. These parameters change each time an image is zoomed or the window is moved. It is not necessary to calculate the parameter for each "zoom" or "move window" operation unless the user identifies a point within the image

3. Convert metes and bounds description into internal system coordinates

4. Convert a point from one coordinate system to another

5. Discard extraneous points between nodes

B. Add, delete, replace or move a spatial entity

This task consists of the following sub-tasks:

1. Add a spatial point, line segment, line or polygon. The spatial entity could be defined by the user or could be created by another Spatial Processing task. This sub-task simply uses Data Base Services to add the specified entity to the Work Space.

2. Delete a portion of a line or polygon

3. Delete a point, line or polygon

4. Move a point and its associated line segments

5. Move a point and a sub-set of its associated line segments

6. Replace points, line segments or polygons

C. Add, delete, replace, or move a text string

This task consists of using Data Base Services to maintain text within a graphics Work Space. It consists of the following subtasks:

1. Maintain the specification for the character size, font and format

2. Maintain the specification for the path of the text. Some examples are:

   - In or offset from a curved line
   - In multiple horizontal or angled lines within a window
   - In a single horizontal or angled straight line

3. Maintain the specification for the location of the first character. Some examples are:

   - At or offset from an absolute point
   - At or offset from a point-type of symbol
   - At or offset from lower/left, center, or upper/right portion of a line

4. Maintain the characters with the text string. For example, change a portion of the text string

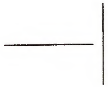
5. Maintain a correlation between the text and its presentation specs. This includes the correlation of a data element with the specs for displaying its value.

D. Move digitized points to the intended location

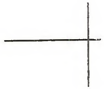
This task consists of accepting digitizing points, correlating them with previously entered points and lines and adjusting, correcting or moving the points to the exact position desired by the user. Some examples are:

1. Move a new point to the closest existing point

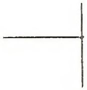
2. Correct the end point of a line that falls short of or over-runs an existing line. Some pictorial examples are:



Short Fall

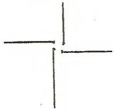


Over Run

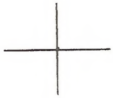


Corrected Position

3. Average the location of groups of points into a single location



Entered Points



Averaged Points

E. Check relationships of two spatial entities

Some examples of this task are:

1. Determine if a point is within a polygon
2. Determine if one line coincides with or intersects another line
3. Determine if a point is on a line segment or polygon perimeter
4. Determine if a polygon intersects with or is contained within another polygon

F. Create spatial entities

This task consists of creating a new spatial entity from a given set of criteria. Some examples are:

1. Create a new polygon whose perimeter is equidistant from a specified point, line or polygon
2. Create a line segment perpindicular to another line at a specified point

3. Create a rectangular parallel to x and y axes, using two diagonal points or at any orientation using these points

4. Create an arc from three unique points

5. Create a circle from three unique points

6. Create the aliquot parts of a township or section

7. Create a new polygon from existing line segments. This includes closing the polygon, if necessary.

8. Create a new polygon from a set of points

9. Consolidate neighboring polygons

10. Create a new set of polygons by merging a new polygon with an existing set of polygons. There are no overlapping polygons within the original set nor within the new set.

11. Create contour lines

12. Consolidate a profile along a vertical plane

G. Calculate dimensions of a spatial entity

Some examples are:

1. Calculate the area of a polygon
2. Calculate the length of a line
3. Calculate the mid-point of a line
4. Calculate the centroid of a polygon
5. Calculate the envelope of a line or polygon

H. Overlay two spatial entities or layers

This consists of the following sub-tasks:

1. Perform the mathematical overlay of two entities or layers in vector form
2. Compute the union or intersection of two overlapping polygons
3. Convert points, lines or polygons to a cellular matrix format

4. Perform cellular overlay of two themes

5. Convert cellular data to vector form

7.5 MATHEMATICS AND STATISTICS

A. Perform mathematic operations

This consists of addition, subtraction, multiplication, division, exponentiation, calculation of roots and logarithmetic functions, etc. on data elements according to parameters which specify the data elements and the operations to be performed on them.

B. Perform standard statistical operations

This consists of calculation of means, standard deviations, variance, etc. on data elements according to parameters which specify the data elements and the operations to be performed on them.

C. Execute user supplied formulas

This consists of weighing, polynomial manipulation, matrix manipulation, time series analysis, interpretation, etc. on data elements according to parameters which specify the data elements and the formulas to be used.

SECTION 8

SYSTEM SUPPORT

8.1 DATA BASE SUPPORT

A. Generate and maintain data base definitions

The data base definitions are generated and maintained by the programs for this task. These data base definitions contain the record names, lengths, keys, where the keys are located in the record, the relationships of records to one another, and the security of the record and/or data base.

B. Generate data bases

The data base is generated using the data base definitions.

C. Perform logging

All changes made to a data base are logged to be used for recovery and backout.

D. Perform backup

The programs for this task will perform backup of data bases that have changes (a dump of the data base).

E. Perform recovery

Recovery of a data base is performed by using the backup of the data base and the log tapes made since the backup.

F. Perform backout

Backout of a data base is performed by using the log tapes to back out changes that have been made.

G. Perform data base reorganization

Reorganization is performed to add and/or delete record types, and to resort physical order of records in a data base,

H. Monitor efficiency of data base access

Monitoring is performed to determine efficiency of usage of the space in the data base and the efficiency of the DBMS of retrieving and updating the data base.

J. Generate and maintain a data dictionary

The programs for this task generate and maintain a data dictionary to be used on-line and in batch mode for verification of alphanumeric and valid ranges, codes, or values. The data dictionary will also be used to determine what records the data element is contained in and its position within the record.

8.2 TELECOMMUNICATIONS SUPPORT

A. Define and maintain telecommunication network

Maintain terminal control and routing table of all terminal devices or system, controlled by user number and terminal ID.

B. Monitor telecommunications activity

Maintain log of terminal usage (correct time) and message log.

C. Assist with isolation of network failure and assist with recovery

This consists of diagnostic and analysis to identify the component which has failed and assisting with swap-over to back-up components if available.

8.3 GENERAL ACTIVITY

A. Perform system accounting

Include all functions to monitor and log computer usage, including periferals. Maintain on-line cost file for all users.

B. Monitor system implementation

Provide management reports on progress of implementation compared to preestablished schedule.

C. Perform system backup

Generate backup types of all disk files in the system, including data files, source and object programs, and job control files.

D. Perform system recovery

In the event of hardware or software malfunction inform users of problem, initiate recovery. If impossible to recover, initiate cold start routines. Maintain log of up and down time.

E. Handle system sign-on and sign-off

Perform checks on user numbers and terminal IDs. Allow no more than three log on attempts. Initiate data and time log for each user. Build I/O device table for each user.

When user terminates session, store data and time of log off and system resource usage for use by system accounting.

F. Handle multi-tasking within system

Allow multiple users to use system simultaneously, including reentrant code.

G. Intercept and process system level abnormal terminations

When program or data errors cause abnormal termination of a program, detect this error and inform user, through error handling of his problem.

H. Process inquiries about system performance and configuration

List location of terminals, their availability and allocated use (includes paired terminals).

SECTION 9

GLOSSARY

GLOSSARY

ACCESS METHOD - a technique for moving data between main storage and devices.

AD HOC - for particular and/or case at hand without consideration of wider application.

ALIQUOT PARTS - legal subdivisions (except fractional lots) or further subdivisions of any legal subdivision, by division into halves or fourths ad infinitum.

ALPHANUMERIC - pertaining to a character set that contains letters, digits, and usually other characters, such as punctuation marks.

ANNOTATION - a note added by way of comment or explanation.

BACKOUT - the process of removing certain changes from a data base by using a log tape of the changes.

BACKUP COPY - a copy of a file, data set, or data base that is kept for reference and/or recovery in case the original is destroyed.

BATCH PROCESSING - pertaining to the technique of executing a set of computer programs such that each is completed before the next program of the set is started.

BATCH JOB - a job that is grouped with other jobs as input to a computing system.

BEST FIT - the resultant of the registration of two spatial entities where the error of registration is averaged over the total set of common control or fit points selected from better entities.

BOUNDARY SEGMENT - set of one or more linked coordinate strings which form a closed area and line segments do not intersect or overlap.

BUFFER - (1) an area of storage that is temporarily reserved for use in performing an input/output operation into which data is read or from which data is written; (2) an area of storage that is temporarily reserved for use of hold data output from a routine and to be used as input to another routine.

CALL - to transfer control to a specified closed subroutine.

CELLULAR MATRIX - a rectangular (or square) array, with entries therein set out in rows, columns, and planes. The individual entries in the arrays are called elements or terms of the matrix. The location of any element in a two- or three-dimensional space can be given by its row, column, and plane location respectively. This simplifies data manipulation since each data location has a unique combination of subscripts to identify it.

CELLULAR OVERLAY - the determination of the logical relationship of two cellular matrices on a cell by cell basis. The two matrice's cell locations must coincide exactly.

CHANGE SCALE - to selectively enlarge or reduce the visual dimensions of a graphic entity while retaining the original dimensions of the graphic textual information associated with the entity.

CHARACTER CODES - the bit configuration assigned to represent a given character.

CLIP - the automated process of shortening an over extended line so that it terminates on a defined boundary.

CODE CONVERSION - a process for changing the bit grouping for a character in one code into the corresponding bit grouping for a character in a second code.

CODED DATA - a set of items, such as abbreviations, representing the members of another set; e.g., (FIPS).

CONNECTOR - a logical operation used to connect statements together.

CONTOUR MAP - a map which portrays a three-dimensional data set by means of contour line. A contour line is an imaginary line on the ground, all points of which have the same value (e.g., elevation, overburden thickness).

CONTROL FLOW - a specification in which programs invoke other programs and in what sequence.

CONTROL POINT - a point on the ground or on a map whose position (horizontal or vertical) is known.

CONVERSATIONAL - pertaining to a program on a system that carries on a dialog with a terminal user, alternately accepting input and then responding to the input quickly enough for the user to maintain his train of thought.

CURSOR - a movable entity either visual (light spot, cross hairs) or physical (digitizer puck), that can be positioned by the user on the display screen or tablets and whose physical location (in inches, master coordinates, etc.) at any given time can be directly retrieved by the software/hardware.

DATA - (1) a representation of facts, concepts, or instructions in a formalized manner suitable for communications, interpretation, or processing by human or automatic means; (2) any representations such as characters or analog quantities to which meaning is, or might be, assigned.

DATA BASE - collection of interrelated data stored together with controlled redundancy to serve one or more applications in an optimal fashion.

DATA BASE DEFINITION - statements used to describe the physical and/or logical structure of the data base.

DATA DICTIONARY - a catalogue of all data types in the data base, giving their name, definitions, and characteristics.

DATA ELEMENT - the smallest unit of data that has meaning to its user.

DATA FLOW - a specification of how data is passed from program to another program.

DATA THEME or OVERLAY - a data base or structure whose graphic entities occupy a unique spatial location (two entities can never cross over one another) and have homogeneous characteristics within a given spatial entity (boundary).

DEVICE - a mechanical, electrical, or electronic contrivance with a specific purpose.

ENTRY - a position within a transaction.

FILE - a collection of related records treated as a unit.

FONTS - a set or assortment of type all of one size and style (e.g., elite or pica).

FUNCTION - a specific purpose of an entity or its characteristic action.

FUNCTION KEY - a terminal key, such as the attention key, that causes the transmission of a signal not associated with a printable character. Detection of the signal usually causes the system to perform some predefined function for the user.

GRAPHICS - the art or science of drawing a visual representation of a spatial entity upon a two-dimensional surface according to mathematical rules of perception.

GRAPHIC PRIMITIVES - device dependent software necessary for the production of graphics on a specific vendor's terminal.

GRID - in optical character recognition, two mutually orthogonal sets of parallel lines used for specifying or measuring character images.

HIERARCHY - a ranked series (e.g., tree structure).

IMAGE - the graphic representation of a mathematically defined spatial entity.

INDEX - an ordered reference list of the contents of a file, data base, or document, together with keys or reference notations for identification or location of those contents.

INTERFACE - a place at which independent systems meet and act on or communicate with each other.

KEY - one or more characters within an item of data that are used to identify it or control its use.

LAYER - pre-defined set of entities relative to a data theme.

LEGEND - an explanatory list of symbols or line types on a map or chart.

MACRO-TRANSACTION - a single transaction that will be expanded to several transactions.

MENU - a list of available entries.

MESSAGE - an arbitrary amount of information whose beginning and end are defined or implied.

METES AND BOUNDS SURVEY - a survey of an irregularly shaped tract of land, not conforming to the rectangular system of surveys.

MODULE - a program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading.

MONITOR - software or hardware that observes, supervises, controls, or verifies the operations of a system.

NODE - the point at which two or more line segments meet. The end point of a line segment.

ON-LINE SYSTEM - in teleprocessing, a system in which the input data enters the computer directly from the point of origin or in which output data is transmitted directly to where it is used.

OPERATOR - in the description of a process, that which indicates the action to be performed on operands.

OUTPUT BUFFER - temporary storage of processed results in device independent format.

POINT - a geometric element defined by a single set of coordinates.

POLYGON - spatial object having a single external boundary, and zero or more internal boundary.

PROGRAM - a series of actions proposed in order to achieve a certain result.

QUALIFIED NAME - a data name explicitly accompanied by specification of the class which belongs in a given classification system.

QUEUE - a waiting line or list formed by items in a system waiting for service.

RANGE - (1) the set of values that a quantity or function may assume; (2) the difference between the highest and lowest value that a quantity or function may assume.

RECORD - a collection of related items of data, treated as a unit.

RECOVERY - the process of using a backup tape and log tapes to restore the data base.

REENTRANT CODE - code that can be used by more than one task at a time.

REGISTRATION - the accurate positioning relative to a reference.

REORGANIZATION - the process of resorting the physical order and location of records in a data base, or adding and/or deleting record types.

ROUTING - the assignment of the communications path by which a message will reach its destination.

SCREEN WINDOW - the viewpoint on a graphics tube in the coordinates of the picture being represented.

SECURITY - prevention of access to or use of data or programs without authorization.

SESSION - the period of time during which a user engages in a dialog with a system.

SIGN-ON - the user-computer interaction required for the computer to recognize the user and to check on any security restrictions.

SPATIAL ENTITY - the numerical representation of a point, line, surface or object.

SYMBOL - a representation of something by reason of relationship, association, or convention. Mnemonic symbol.

SYNTAX - (1) the structure of expressions in a language; (2) the rules governing the structure of a language.

TELECOMMUNICATIONS - data transmission between a computing system and remotely located devices via a unit that performs the necessary format conversion and controls the rate of transmission.

TERMINAL - a device, usually equipped with a keyboard and some kind of display, capable of sending and receiving information over a communication channel.

THEME - multiple attributes, spatial entities must be mutually exclusive.

TRANSACTION - represents the system's view of a complete user input - consists of instructions and data.

VECTOR - a line segment comprised of two end points.

VIEWABLE PORTION - that portion of a spatial entity that can be seen by the user.

ZOOM IN/OUT - to increase or decrease the scale or size of a graphic image.

BLM LIBRARY
RS 150A BLDG. 50
DENVER FEDERAL CENTER
P.O. BOX 25047
DENVER, CO 80225

QA 76.8 .B85 C663 1977 v.2 c.2
Conceptual system structure
for BLM REsource

BLM LIBRARY
RS 150A BLDG. 50
DENVER FEDERAL CENTER
P.O. BOX 25047
DENVER, CO 80225

BLM LIBRARY
SC-653, BLDG. 50
DENVER FEDERAL CENTER
P. O. BOX 25047
DENVER, CO 80225-0047